

Revisiting Gradient Pruning: A Dual Realization for Defending against Gradient Attacks

Lulu Xue¹, Shengshan Hu^{1*}, Ruizhi Zhao¹, Leo Yu Zhang², Shengqing Hu³, Lichao Sun⁴, Dezhong Yao⁵

¹ School of Cyber Science and Engineering, Huazhong University of Science and Technology

² School of Information and Communication Technology, Griffith University

³ Department of Nuclear Medicine, Union Hospital, Tongji Medical College, Huazhong University of Science and Technology

⁴ Department of Computer Science and Engineering, Lehigh University

⁵ School of Computer Science and Technology, Huazhong University of Science and Technology

{lluxue,hushengshan,zhaorui,dyao}@hust.edu.cn, hsqha@126.com, leo.zhang@griffith.edu.au, james.lichao.sun@gmail.com

Abstract

Collaborative learning (CL) is a distributed learning framework that aims to protect user privacy by allowing users to jointly train a model by sharing their gradient updates only. However, gradient inversion attacks (GIAs), which recover users' training data from shared gradients, impose severe privacy threats to CL. Existing defense methods adopt different techniques, e.g., differential privacy, cryptography, and perturbation defenses, to defend against the GIAs. Nevertheless, all current defense methods suffer from a poor trade-off between privacy, utility, and efficiency. To mitigate the weaknesses of existing solutions, we propose a novel defense method, Dual Gradient Pruning (DGP), based on gradient pruning, which can improve communication efficiency while preserving the utility and privacy of CL. Specifically, DGP slightly changes gradient pruning with a stronger privacy guarantee. And DGP can also significantly improve communication efficiency with a theoretical analysis of its convergence and generalization. Our extensive experiments show that DGP can effectively defend against the most powerful GIAs and reduce the communication cost without sacrificing the model's utility.

1 Introduction

Collaborative learning (CL) (Shokri and Shmatikov 2015) is a distributed learning framework, where multiple users train a model locally and share their gradients among the peers or to a centralized server. CL claims to protect user privacy since users do not need to share their local (private) data directly. However, recent studies reveal that gradients can be used to recover the original training data information via gradient inversion attacks (GIAs) (Zhu, Liu, and Han 2019; Geiping et al. 2020). To against GIAs, a large number of studies have been proposed, where they leverage the advanced privacy protection techniques, such as differential privacy (DP) (Dwork, Roth et al. 2014), cryptography (Bonawitz et al. 2017; Hardy et al. 2017; Gilad-Bachrach et al. 2019) and perturbation defense (Gao et al. 2021; Sun et al. 2021; Scheliga, Mäder, and Seeland 2022).

However, none of the existing defense methods could take care of all privacy, utility, and efficiency difficulties in the CL framework.

For example, traditional defenses such as DP and cryptography-based methods strike a balance among privacy protection, model performance, and efficiency simultaneously (Dwork, Roth et al. 2014; Bonawitz et al. 2017; Hardy et al. 2017; Gilad-Bachrach et al. 2019). To address this challenge, various perturbations-based methods have been proposed (Gao et al. 2021; Sun et al. 2021; Scheliga, Mäder, and Seeland 2022). But they all rely on auxiliary optimization modules to reduce certain privacy leakage and cannot defend against all GIAs in practice (see Sec. 6.2 for details). For instance, perturbation-based defense methods (*i.e.*, Precode (Scheliga, Mäder, and Seeland 2022), Soteria (Sun et al. 2021)) can effectively defend against passive GIAs (Geiping et al. 2020; Wang et al. 2020; Wei et al. 2020b), but fail to work against the active GIAs (Boenisch et al. 2021; Pan et al. 2022), which is considered as the state-of-the-art attack method. On the contrary, the classic Top- k based gradient pruning method (Lin et al. 2017; Alistarh et al. 2018) is generally ineffective for enhancing privacy against passive GIAs, and corresponding defenses (*e.g.*, Outpost (Wang, Hugh, and Li 2023)) offer limited protection. But we find that they significantly outperform recent defense methods under the active attack. Tab. 1 gives a detailed experimental result for this observation. The new findings inspire us to seek a more practical and effective defense against both passive and active GIAs. In this paper, we propose a new gradient pruning-based method, **Dual Gradient Pruning (DGP)**. Dual gradient pruning is a novel gradient pruning technique, which removes top- k_1 largest gradient parameters and the bottom- k_2 smallest gradient parameters from the local model. DGP leads to a strong privacy protection against both passive GIAs and active GIAs.

To measure the level of protection, we present the theoretical analysis of reconstruction error from pruned gradients, showing that the error is proportional to gradient distance. So removing larger gradient parameters can rapidly enlarge the gradient distance, resulting in a significant reconstruction error. However, removing many larger parameters will

*Corresponding Author.

significantly impact the model’s utility. Thus, to improve the pruning ratio, which is essential to robustness against active attack (Boenisch et al. 2021; Fowl et al. 2021), we also remove smaller gradient parameters. In this way, our method could significantly mitigate GIAs without affecting the model’s utility.

We conduct extensive experiments to evaluate our method. The quantitative and visualized results show that our design can effectively make recovered images unrecognizable under different attacks, and reduce the communication cost. Our contributions are as follows: 1) We revisit gradient pruning to show its potential for mitigating GIAs; 2) We propose an improved gradient pruning strategy to provide sufficient privacy guarantee while balancing the model accuracy and the system efficiency; 3) We conduct extensive experiments to show that our design outperforms existing defense methods *w.r.t.* privacy protection, model accuracy, and system efficiency.

2 Related Work

Collaborative learning (Shokri and Shmatikov 2015) is considered to be a privacy-preserving framework for distributed machine learning as the training data is not directly outsourced. However, the emerging of GIAs (Zhu, Liu, and Han 2019; Fan et al. 2020; Zhao, Mopuri, and Bilen 2020; Geiping et al. 2020; Qian and Hansen 2020; Boenisch et al. 2021; Yin et al. 2021; Zhu and Blaschko 2020; Fowl et al. 2021) shatters this conception. It has been proven that the attacker (*e.g.*, a curious server) can easily recover the private data from gradient to a great extent. The privacy guarantee of collaborative learning urgently needs to be strengthened.

Traditional Defense. Traditionally, there are two approaches to construct privacy-preserving collaborative learning: using DP to disturb gradients (Dwork, Roth et al. 2014; Abadi et al. 2016; Geyer, Klein, and Nabi 2017; Yu et al. 2019; Chen, Wu, and Hong 2020) or using cryptographic tools to perform secure aggregation (Danner and Jelasity 2015; Bonawitz et al. 2017; Hardy et al. 2017; Mohassel and Zhang 2017; Sun, Qian, and Chen 2021; Gilad-Bachrach et al. 2019). DP (Dwork, Roth et al. 2014) is a popular and effective privacy protection mechanism by adding random noise to the raw data, but it is well known that the noise introduced by DP can greatly degrade the model accuracy when meaningful privacy is enforced (Wei et al. 2020a). Cryptographic-based secure aggregation can guarantee both privacy and accuracy simultaneously, but it incurs expensive computation and communication costs (Kairouz et al. 2021). Using the shuffle model (Liu et al. 2020; Sun, Qian, and Chen 2021) can only provide anonymity. Moreover, it totally changes the system model of collaborative learning since an additional semi-trusted third party is introduced to work cooperatively with the server.

Perturbation Defense. Recently, researchers have begun to explore the possibility of constructing new gradient perturbation mechanisms to better balance privacy and accuracy. (Sun et al. 2021) proposed Soteria, a scheme that perturbs the representation of inputs by pruning the gradients of a single layer. (Gao et al. 2021) proposed ATS, an optimized training data augmentation policy by transforming

original sensitive images into alternative inputs, to reduce the visibility of reconstructed images. (Scheliga, Mäder, and Seeland 2022) presented Precode to extend the model architecture by using variational bottleneck (VB) (Alemi et al. 2016) to prevent attackers from obtaining optimal solutions to reconstructed data. These works focus on the semi-honest setting (Zhu, Liu, and Han 2019; Wang et al. 2019; Wei et al. 2020b) but fail to protect privacy when an active server modifies the model to launch GIAs (Fowl et al. 2021). Moreover, these works suffer from high computation costs or a huge communication burden.

Gradient Pruning Defense. From an independent research domain, gradient pruning has been commonly used for saving communication bandwidth. The most common pruning strategy is Top- k selection, which retains top k gradient parameters with the largest absolute values (Lin et al. 2017; Alistarh et al. 2018). It has been widely proved that gradient pruning provides very limited privacy protection ability (Zhu, Liu, and Han 2019; Gao et al. 2021; Huang et al. 2021; Sun et al. 2021; Scheliga, Mäder, and Seeland 2022) unless a high pruning ratio (*e.g.*, removing 99% of the gradients) is used at the cost of 10% accuracy drop (Huang et al. 2021). However, we emphasize that this is misunderstood as they only consider the Top- k selection strategy and it has never received an in-depth investigation in the field of security. It is originally designed for improving system efficiency, thus a direct application inherently suffers from many weaknesses. Recently, (Wang, Hugh, and Li 2023) proposed Outpost, a privacy-preserving method that combines Top- k gradient pruning with adaptive noise addition. However, our experiments indicate that Outpost cannot effectively defend against passive GIAs. In contrast, our work shows that a slight modification can unleash the potential of gradient pruning to provide a strong privacy guarantee, as shown in Sec. 4.

3 Threat Model and Gradient Attacks

In this work, we consider a strong threat scenario, where an active server, after receiving gradients from users, tries to reconstruct the local training data and is motivated to modify model parameters in each iteration to strengthen the attack effect. As will be shown in Sec. 5 and Sec. 6, our method provides a theoretical guarantee against passive attacks and empirical protection against active attacks. So we briefly discuss both kinds of attacks below.

Analytical Attack (Passive). Analytical attack exploits the structure of the gradients to recover the inputs, such as using gradient bias terms (Phong et al. 2017). Recently proposed R-gap attack (Zhu and Blaschko 2020) exploits the recursive relationship between gradient layers to solve the input. An effective analytical attack depends on the specific structure and parameters of gradients.

Optimization Attack (Passive). Optimization attack is firstly proposed in (Zhu, Liu, and Han 2019), which approximates the desired data (\mathbf{x}, \mathbf{y}) with dummy data $(\mathbf{x}^*, \mathbf{y}^*)$ by optimizing the euclidean distance between the gradients \mathbf{g}^* (generated by dummy data $(\mathbf{x}^*, \mathbf{y}^*)$) and the original gradients $\nabla \mathbf{W}$ (produced by real private data (\mathbf{x}, \mathbf{y})) with L-BFGS optimizer. (Geiping et al. 2020) proposed IG, opti-

mizing the cosine distance with Adam optimizer, and (Yin et al. 2021) proposed GI, optimizing the Euclidean distance with Adam optimizer. These methods are state-of-the-art optimization attacks. Furthermore, recent works (Yin et al. 2020; Li et al. 2022) utilize GANs to generate data approximating the input. However, these attacks are impractical as they necessitate training GANs with vast amounts of data that closely resemble private data.

Despite different optimizers can be used to achieve better attack quality (Geiping et al. 2020; Wang et al. 2020; Wei et al. 2020b), the existing attacks are all measured by the distance between the virtual gradients \mathbf{g}^* and the original gradients $\nabla \mathbf{W}$. We therefore propose a general definition for passive attacks to better evaluate their performance. From Definition 1, for a given success probability $(1 - \delta)$, a smaller ε indicates a better attack strategy \mathcal{A} .

Definition 1. A passive attack \mathcal{A} is a (ε, δ) -passive attack, if it satisfies:

$$\mathbb{P}(\mathbb{E}(\mathcal{D}_{\mathcal{A}}(\nabla \mathbf{W}, \mathbf{g}^*)) \leq \varepsilon) \geq 1 - \delta. \quad (1)$$

where \mathbb{P} represents the probability, \mathbb{E} represents the expectation, $\mathcal{D}_{\mathcal{A}}$ is the distance (commonly instantiated with Euclidean or cosine distance) estimated under \mathcal{A} .

Active Server Attack. In this kind of attack, the server can actively modify the global model to realize a better attack result rather than honestly executing the protocols (Boenisch et al. 2021; Pan et al. 2022; Wen et al. 2022). Recently proposed Rob attack (Fowl et al. 2021) adds imprint modules to the model and uses the difference between the gradient parameters in adjacent rows of the imprint module to recover the data, achieving the best attack effect in the literature.

4 Dual Gradient Pruning

4.1 Analysis of Gradient Pruning

We owe the failure of common Top- k gradient selection methods to two reasons: 1) the distance between the Top- k pruned gradient \mathbf{g} and the real gradient $\nabla \mathbf{W}$ is small; and 2) large gradient parameters in $\nabla \mathbf{W}$ also reveal label information about user data. The first reason stems from the intuitive observation that when the perturbed gradient is close to the true gradient, it becomes easier for the attacker to infer sensitive information about the true gradient. And we give a specific example to illustrate this point. In particular, Fig. 1 plots the recovery results of IG attack (in terms of PSNR (\downarrow), MSE (\uparrow), LPIPS (Zhang et al. 2018) (\uparrow), SSIM (Wang et al. 2004) (\downarrow) metrics) under various relative gradient distance $\|\nabla \mathbf{W} - \mathbf{g}\|_2 / \|\nabla \mathbf{W}\|_2$ (measured in ratio). It is clear from the figure that greater distance leads to worse reconstruction for all metrics. To better support this observation, we propose the following non-rigorous proposition.

Proposition 1. For any given input \mathbf{x} and shared model \mathbf{W} , the distance between the recovered data \mathbf{x}' and the real data \mathbf{x} is bounded by:

$$\|\mathbf{x} - \mathbf{x}'\|_2 \geq \frac{\|\varphi(\mathbf{x}, \mathbf{W}) - \varphi(\mathbf{x}', \mathbf{W})\|_2}{\|\partial \varphi(\mathbf{x}, \mathbf{W}) / \partial \mathbf{x}\|_2}, \quad (2)$$

where φ is the mapping from input to the gradient, i.e., the reconstruction quality is limited by $\|\varphi(\mathbf{x}, \mathbf{W}) - \varphi(\mathbf{x}', \mathbf{W})\|_2 = \|\nabla \mathbf{W} - \mathbf{g}\|_2$.

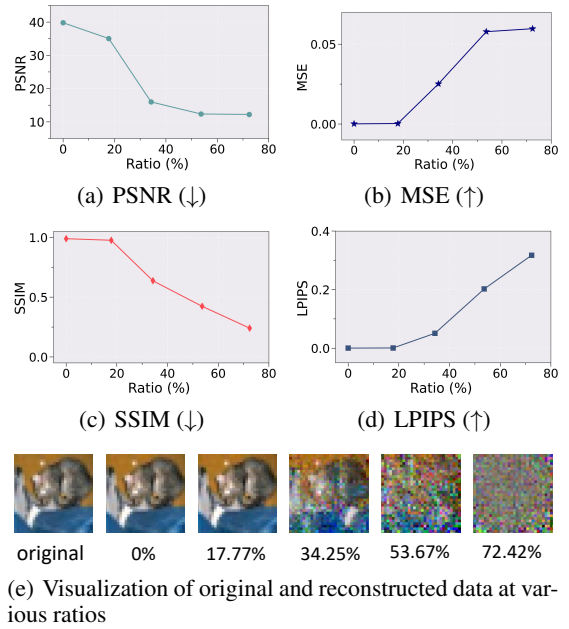


Figure 1: Relationship between relative gradient distance and reconstruction quality under IG (CIFAR10(Krizhevsky, Hinton et al. 2009) with ResNet18 (He et al. 2016)).

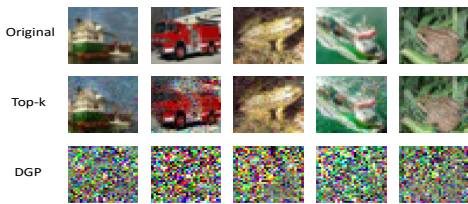
Referring to the proof technique of Lemma 1 in (Sun et al. 2021), we employ the first-order Taylor expansion in our proof. The specific proof of the above proposition is moved to the appendix due to space limit (the same hereinafter). And we will present a more rigorous analysis in our follow-up study. According to the above example and this proposition, it is clear that the reconstruction error is proportional to the gradient distance $\|\nabla \mathbf{W} - \mathbf{g}\|_2$, i.e., effective defense methods should enlarge the gradient distance as much as possible. However, for the Top- k gradient selection (Lin et al. 2017; Alistarh et al. 2018), the k largest parameters are retained, making the gradient distance small by nature. To explain the second reason, we consider a L -layer perceptron model trained with cross-entropy loss for classification. Let a column vector $\mathbf{r} = [r_1, r_2, \dots, r_n]$ be the logits (the output of the L -th linear layer) that input to the softmax layer, the confidence score probability vector is thus $\left[\frac{e^{r_1}}{\sum_j e^{r_j}}, \frac{e^{r_2}}{\sum_j e^{r_j}}, \dots, \frac{e^{r_n}}{\sum_j e^{r_j}} \right]$ and the succinct form of the cross-entropy loss becomes $\ell(\mathbf{x}, y) = -\log\left(\frac{e^{r_y}}{\sum_j e^{r_j}}\right)$.

Focus on the L -th layer $\mathbf{W}^L \mathbf{x} + \mathbf{b}^L = \mathbf{r}$, it is easy to find $\frac{\partial \ell(\mathbf{x}, y)}{\partial b_i} = \frac{\partial \ell(\mathbf{x}, y)}{\partial r_i} \cdot \frac{\partial r_i}{\partial b_i} = \frac{\partial \ell(\mathbf{x}, y)}{\partial r_i} = \frac{e^{r_i}}{\sum_j e^{r_j}} - \mathbb{I}_{i=y}$,

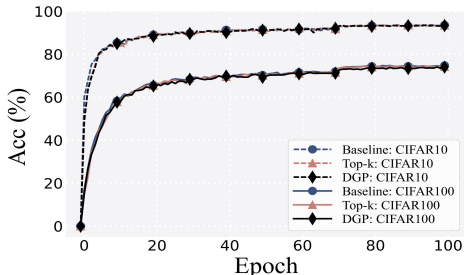
and

$$\nabla \mathbf{W}^L = \frac{\partial \ell(\mathbf{x}, y)}{\partial \mathbf{r}} \cdot \mathbf{x}^T = \left[\frac{\partial \ell(\mathbf{x}, y)}{\partial r_1}, \dots, \frac{\partial \ell(\mathbf{x}, y)}{\partial r_n} \right] \cdot \mathbf{x}^T.$$

For a given \mathbf{x} (and so \mathbf{x}_t is fixed), the magnitude of certain elements of the gradient matrix $\nabla \mathbf{W}^L$ (i.e., the i -th row) is particularly large if i is the true label of the training data \mathbf{x} due to reason that $\left| \frac{\partial \ell(\mathbf{x}, y)}{\partial r_i} \right| = \sum_{j \neq i} \left| \frac{\partial \ell(\mathbf{x}, y)}{\partial r_j} \right|$.



(a) Recovered data under IG



(b) ResNet18 on CIFAR dataset

Figure 2: Comparison between Top- k and DGP on privacy and accuracy (20% of parameters are selected in Top- k).

To summarize, due to the above two reasons, we conclude that common Top- k gradient selection cannot provide sufficient protection for user data against passive optimization attacks. From another point of view, a sufficient gradient pruning ratio also plays an important role in defending against active server attacks. As mentioned in Sec. 3, active attackers can exploit the correspondence of partial gradient parameters to recover the real data. So, the gradient pruning will directly destroy the relationship among gradient parameters constructed by the active attacker. Intuitively, the higher the pruning rate, the stronger the impact. As will be validated in Sec.6, a high pruning rate can prevent the attacker from obtaining useful gradient information.

4.2 Dual Gradient Pruning

Generally speaking, large gradient parameters of local model need to be removed to make the gradient distance larger, but the distance should also be appropriately bounded to maintain high model accuracy. Moreover, it is also necessary to delete gradient parameters to achieve a high pruning ratio, which can reduce the input information that the active server may retain on the gradient by modifying the model and improve communication efficiency. Considering the model performance, we choose to remove small gradient parameters to achieve this. With these observations, we propose dual gradients pruning (DGP), a new parameter selection strategy for gradient pruning. The users first layer-wisely sort the absolute values of local gradient parameters $\nabla \mathbf{W}$ in the descending order. Let $\mathcal{T}_{k_1}(\nabla \mathbf{W})$ represent the set of top- k_1 percents of elements of $\nabla \mathbf{W}$, $\mathcal{B}_{k_2}(\nabla \mathbf{W})$ represent the set of its bottom- k_2 percents. Then the users remove $\mathcal{T}_{k_1}(\nabla \mathbf{W})$ and $\mathcal{B}_{k_2}(\nabla \mathbf{W})$ from $\nabla \mathbf{W}$ for gradient pruning. A detailed illustration of DGP is shown in Alg. 1. Note that we set $p = k_1/k_2$ as a hyperparameter to regu-

Algorithm 1: Dual Gradient Pruning (DGP).

Require:

- Original gradient matrix $\nabla \mathbf{W}$, values of k_1 and k_2 .
 - 1: **for** $l \leftarrow 1$ to L **do**
 - 2: Search sets $\mathcal{T}_{k_1}(\nabla \mathbf{W}^l)$ and $\mathcal{B}_{k_2}(\nabla \mathbf{W}^l)$.
 - 3: Obtain \mathbf{g}^l by removing the parameters in $\mathcal{T}_{k_1}(\nabla \mathbf{W}^l)$ and $\mathcal{B}_{k_2}(\nabla \mathbf{W}^l)$ from $\nabla \mathbf{W}^l$.
 - 4: **end for**
 - 5: return Pruned gradient matrix $\mathbf{g} = \{\mathbf{g}^i\}_{i=1}^L$.
-

Algorithm 2: A Complete Illustration of Our Defense.

Require:

- Initial model \mathbf{W}_0 , value k_1 and k_2 , total rounds T , total users N .
 - 1: Set $\mathbf{e}_0 = 0$.
 - 2: **for** $t \leftarrow 0$ to $T - 1$ **do**
 - 3: **for** $i \leftarrow 1$ to N **do**
 - 4: The i -th user generates local gradient $\nabla \mathbf{W}_{t,i}$.
 - 5: $\mathbf{P}_{t,i} = \nabla \mathbf{W}_{t,i} + \mathbf{e}_{t,i}$.
 - 6: $\mathbf{g}_{t,i} = \text{DGP}(k_1, k_2, \mathbf{P}_{t,i})$
 - 7: $\mathbf{e}_{t+1,i} = \mathbf{P}_{t,i} - \mathbf{g}_{t,i}$
 - 8: **end for**
 - 9: Server side aggregation:
 - 10: $\mathbf{W}_{t+1} = \mathbf{W}_t - \eta \frac{\sum_{i=1}^N \mathbf{g}_{t,i}}{N}$
 - 11: **end for**
 - 12: return Shared global model \mathbf{W}_T .
-

late the trade-off between privacy and accuracy. The authors in (Lin et al. 2017) show that large gradient parameters are more likely to have an impact on the model’s performance, hence removing these large parameters will reduce model’s accuracy. To reduce this negative impact and increase convergence speed, we introduce the error feedback mechanism (Karimireddy et al. 2019). In particular, at the iteration round t , after user i obtaining his local gradient $\nabla \mathbf{W}_{t,i}$, he will combine $\nabla \mathbf{W}_{t,i}$ with an error term accumulated in the previous $(t - 1)$ rounds before performing the DGP. A complete illustration of our method is shown in Alg. 2, and the steps from $\mathbf{e}_{t,i}$ to $\mathbf{e}_{t+1,i}$ provide the implementation details of error feedback mechanism. We emphasize that although such dual gradients pruning strategy is very simple, it can significantly mitigate GIAs without affecting the model accuracy. Fig. 2(a) gives an example of ResNet18 showing the privacy guarantee when $k_1 = 5\%$, $k_2 = 75\%$. Fig. 2(b) gives a comparison of model performance. The convergence analysis of our method is shown in Sec. 5, and more experimental results can be found in Sec. 6.

5 Theoretical Analysis

This section presents the security analysis with regard to passive GIAs, as well as the generalization and convergence analyses of our method.

5.1 Assumptions

Following the literature studies in (Wilson et al. 2017; Karimireddy et al. 2019), for a given L -layer centralized model, we model the first $(L - 1)$ layers as a robust feature extractor of any input sample. Thus, the function of this model is characterized by $f(x|\mathbf{W}) = \mathbf{W}x + \mathbf{b}$, and the optimization objective is the loss $\ell(\mathbf{x}, y)$ (such as cross-entropy). To facilitate analyses and following literature studies (Chen et al. 2020; Dai et al. 2019; Karimireddy et al. 2019), the assumptions about the smoothness of DGP and l , as well as the variance of the stochastic gradient are employed.

Assumption 1. *The pruning mechanism $\text{DGP}(k_1, k_2, \cdot)$ is Lipschitz, so the following conditions hold:*

$$\begin{aligned} & \|\nabla \mathbf{W} - \text{DGP}(k_1, k_2, \nabla \mathbf{W})\|_2^2 \\ &= \|\text{DGP}(0, 0, \nabla \mathbf{W}) - \text{DGP}(k_1, k_2, \nabla \mathbf{W})\|_2^2 \leq \gamma_1 \|\nabla \mathbf{W}\|_2^2, \end{aligned}$$

where γ_1 is a constant related to k_1 and k_2 and satisfies $(1 - \sqrt{1 - k_1 * k_2})^2 < \gamma_1 < 1$.

Assumption 2. *The objective function $l : \mathbb{R}^d \rightarrow \mathbb{R}$ has a low bound l^* and it is Lipschitz-smooth, i.e., for any x_1, x_2 , $\|\nabla l(x_1) - \nabla l(x_2)\|_2 \leq K\|x_1 - x_2\|_2$ and $l(x_1) \leq l(x_2) + \langle \nabla l(x_2), x_1 - x_2 \rangle + \frac{K}{2}\|x_1 - x_2\|_2^2$.*

Assumption 3. *The collaborative stochastic gradient $\nabla \mathbf{W}_{t,i}$ ($t = [0, T - 1], i = [1, N]$) is bounded, i.e., $\|\nabla \mathbf{W}_{t,i}\|_2^2 \leq G^2$, and the average aggregated gradient $\nabla \mathbf{W}_t$ is the expectation of collaborative stochastic gradient $\nabla \mathbf{W}_{t,i}$, i.e., $\nabla \mathbf{W}_t = \mathbb{E}(\nabla \mathbf{W}_{t,i})$. Moreover, the variance between $\nabla \mathbf{W}_{t,i}$ and $\nabla \mathbf{W}_t$ is bounded: $\mathbb{E}\|\nabla \mathbf{W}_{t,i} - \nabla \mathbf{W}_t\|_2^2 \leq \sigma^2$.*

5.2 Security Analysis

When considering passive attacks, we prove that DGP achieves a stronger privacy protection in the sense of Definition 1.

Theorem 1. *For any (ε, δ) -passive attack \mathcal{A} , under the presence of DGP, it will be degenerated to $(\varepsilon + \sqrt{\gamma_1}\|\nabla \mathbf{W}\|_2, \delta)$ -passive attack if $\mathcal{D}_{\mathcal{A}}$ is measured by Euclidean distance, and degenerated to $(\varepsilon + (1 - \varepsilon)\sqrt{\gamma_1}, \delta)$ -passive attack if $\mathcal{D}_{\mathcal{A}}$ is measured by cosine distance.*

Theorem 3 is based on Assumption 1 about DGP. It reveals that, with the same successful chance $(1 - \delta)$, DGP weakens the passive attack \mathcal{A} 's capability to obtain a better estimation of the true $\nabla \mathbf{W}$. In particular, \mathcal{A} 's estimation of $\nabla \mathbf{W}$ is enlarged by $\sqrt{\gamma_1}\|\nabla \mathbf{W}\|_2$ under Euclidean distance and enlarged by $(1 - \varepsilon)\sqrt{\gamma_1}$ under cosine distance.

5.3 Convergence Guarantee

We start the convergence analysis by proving the generalization of DGP. The generalization analysis aims to quantify how the trained model performs on the test data, and it is achieved by analyzing the how DGP affects the properties of the optima reached (without gradient pruning) (Karimireddy et al. 2019; Wilson et al. 2017). For ease of expression, let CL-SGD represent the training in CL with the SGD optimizer. Based on Assumptions 1 and 3, the following Lemma can be obtained.

Lemma 1. *Let $\mathbf{e}_t = \sum_{i=1}^N \mathbf{e}_{t,i}/N$ be the averaged accumulated error among all users at iteration t , the expectation of the norm of \mathbf{e}_t is bounded, i.e.,*

$$\mathbb{E}\|\mathbf{e}_t\|_2^2 \leq \frac{3\gamma_1(2 + \gamma_1)}{2(1 - \gamma_1)^2} G^2. \quad (3)$$

Note that the difference between the averaged pruned gradient $\mathbf{g}_t = \sum_{i=1}^N \mathbf{g}_{t,i}/N$ and the averaged collaborative SGD gradient $\nabla \mathbf{W}_t = \sum_{i=1}^N \nabla \mathbf{W}_{t,i}/N$ is simply $\|\sum_{i=0}^{T-1} (\nabla \mathbf{W}_t - \mathbf{g}_t)\|_2^2 = \|\mathbf{e}_T\|_2^2$. So the lemma above indicates that the accumulated gradient difference between our algorithm and CL-SGD is bounded. That said, the optima reached by DGP and the optima reached by CL-SGD will eventually be very close if the algorithm converge. Armed with Lemma 2 and based on Assumptions 1, 2 and 3, we demonstrate the convergence of the our algorithm.

Theorem 2. *The averaged norm of the full gradient $\nabla l(\mathbf{W}_t)$ derived from centralized training is correlated with the our algorithm as follows:*

$$\begin{aligned} \frac{\sum_{t=0}^{T-1} \mathbb{E}\|\nabla l(\mathbf{W}_t)\|_2^2}{T} &\leq 4 \frac{l^0 - l^*}{\eta T} + 2K\eta(G^2 + \sigma^2) \\ &\quad + 4\eta^2 K^2 \frac{3\gamma_1(2 + \gamma_1)}{2(1 - \gamma_1)^2} G^2, \end{aligned} \quad (4)$$

where l^0 is the initialization of l , and η is the learning rate.

The implication of Theorem 4 is that, with an appropriate learning rate η , DGP converges similar to CL-SGD (slower by a negligible term $\mathcal{O}(\frac{1}{\sqrt{T}})$), as shown in Corollary 1.

Corollary 1. *Let $\eta = (l^0 - l^*)/KT(G^2 + \sigma^2)$, we have*

$$\begin{aligned} \frac{\sum_{t=0}^{T-1} \mathbb{E}\|\nabla l(\mathbf{W}_t)\|_2^2}{T} &\leq 6\sqrt{\frac{K(l^0 - l^*)(\sigma^2 + G^2)}{T}} \\ &\quad + \mathcal{O}\left(\frac{1}{T}\right). \end{aligned}$$

6 Experiments

6.1 Experimental Setup

We run the experiments with PyTorch by using one RTX 2080 Ti GPU and a 2.10 GHz CPU. For fair comparison, we follow the setting of (Gao et al. 2021), using ten users with the same data distribution. We assess model privacy against various attacks and evaluate model performance on CIFAR10 and CIFAR100, which is a common setting used in many studies (Huang et al. 2021; Gao et al. 2021). We follow (Huang et al. 2021; Jeon et al. 2021) to quantify the privacy effect of defenses, i.e., visualizing the reconstructed data and using learned perceptual image patch similarity (LPIPS) and structural similarity (SSIM) to measure the quality of the recovered data. A better defense should have larger LPIPS (\uparrow) and smaller SSIM (\downarrow).

Attack methods. We evaluate DGP against IG, GI, R-gap, and Rob attacks, which represent state-of-the-art passive and active GIAs, as discussed in Sec. 3. We use the following default attack settings: ResNet18 for IG, GI, Rob on CIFAR10. And we apply R-gap with CNN6 (Zhu and Blaschko

2020) on CIFAR10, as this analysis attack is only suitable for models with simple structures. We provide additional attack details, more privacy evaluations (*e.g.* more models and datasets) and efficiency evaluation (computation costs and communication costs) in the appendix.

Defense methods. We compare DGP with six state-of-the-art defenses: Soteria, ATS, Precode, Outpost, DP and Top- k pruning. Besides, we set CL-SGD as the baseline that adopts no defense. Note that DP provides privacy guarantee by adding noise to gradients in deep learning. We adhere to the DP settings of (Sun et al. 2021) and use Gaussian noise with standard deviation $\sigma = 10^{-2}$. When quantifying the defense performance of ATS, we not only evaluate the similarity between the raw images and the recovered data (ATS-T), but also evaluate the similarity between the disturbed training images (*i.e.*, the real inputs) and the recovered data (ATS-R). For Top- k and DGP, we set $k = 20\%$, $k_1 + k_2 = 80\%$ with the regulation hyperparameter $p = 1/15$. The rest defenses remain the original settings.

6.2 Privacy Evaluation

Tab. 1 shows the defense performance with SSIM, and LPIPS under four attacks. For each metric, we bold the best result and underline the second best result (the same hereinafter). The results show that ATS, Soteria, Precode, DP perform poorly under Rob attack, while Top- k and Outpost are vulnerable to IG attack and GI attack. In summary, DGP can provide excellent privacy protection under all attacks, while still retain high model accuracy. To perceptually demonstrate the defense performance, we also visualize the reconstructed images. Note that ATS-T refers to processed raw data, while ATS-R represents the reconstructed raw data in Fig. 3. Fig. 3(a) and Fig. 3(b) depict the recovered images under optimization attacks (*e.g.*, IG, GI). We can find that the attacker can still recover the outline of inputs with ATS, Top- k and Outpost. Soteria, Precode, DP and DGP can make the recovered images unrecognizable. Fig. 3(c) shows the recovered images from the R-gap attack. We can see that all defenses but ATS can well defend against R-gap because ATS does not damage the gradient structure, validating that a slight perturbation on gradients can mitigate the analytical attacks easily. We are not able to provide the result of Precode because its VB operation destroys the model structure, making R-gap cannot be mounted. Fig. 3(d) plots the recovered images from the Rob attack. It shows that ATS, Precode, and Soteria fail to work and most inputs can be reconstructed. Fig. 3(d) shows that DP also cannot defend against Rob. This might be because the server calculates the inputs by superimposing a large number of the malicious imprint module’s gradient parameters. And the noise added to the gradient follows a normal distribution, potentially canceling out when aggregated in large numbers. However, DGP, Top- k , and Outposts can effectively defend against Rob attack because the gradients of all layers are pruned, including those of the malicious imprint modules. However, we reiterate that the main weakness of the gradient pruning based on Top- k selection is its vulnerability to optimization attacks (*e.g.*, IG, GI), as widely demonstrated in the literature (Gao et al. 2021; Sun et al. 2021).

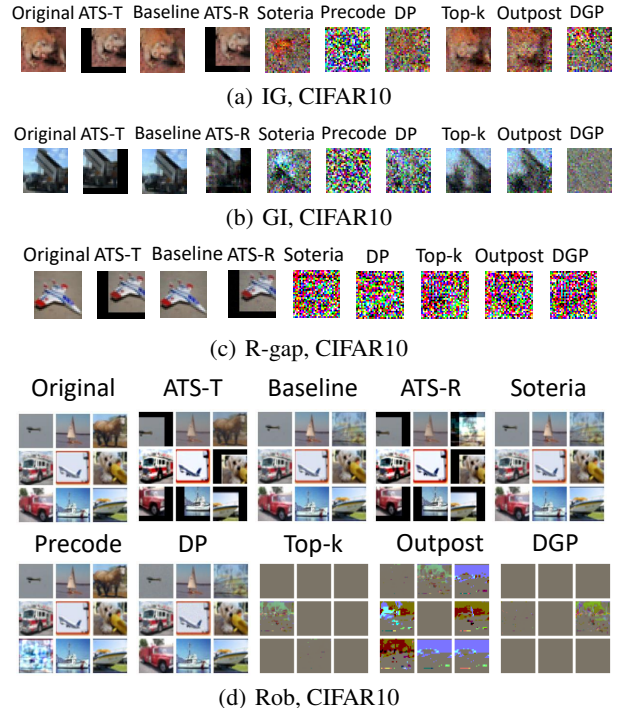


Figure 3: Data visualization on privacy evaluation by using multiple gradient inversion attacks.

6.3 Accuracy Evaluation

Tab. 1 lists the accuracy of ResNet18 on CIFAR10 under different defenses. Clearly, ATS, Soteria, Precode, Outpost, Top- k and our method can achieve model accuracy similar to the unprotected baseline, while DP performs worst as expected. Additionally, we evaluated more model performance with DGP, including ResNet18, VGG11 (Simonyan and Zisserman 2014), CNN6, LeNet (Zhu) (Geiping et al. 2020). And we further perform ablation experiments to explore the role of the error feedback mechanism. Fig. 4 shows that the model performance of DGP with error feedback is close to the baseline. However, DGP without error feedback performs poorly and even fails to converge. This is because accumulated errors result in a larger disparity between the model’s update direction and the correct update direction. Notably, this effect is mitigated in structurally complex models due to the presence of numerous redundant parameters. Prior research (Molchanov et al. 2016) indicated that even if these redundant parameters are not updated (*i.e.*, their gradient parameters are set to 0), their impact on model performance is small. Our theoretical analysis and Fig. 4 show that the error feedback mechanism can effectively correct the negative effects caused by gradient pruning. And Top- k method can also enjoy the benefit since it is also based on pruning. However, further experiments (see details in the appendix) validate that, to achieve a similar level of privacy protection of DGP with 80% pruning, the pruning rate of Top- k exceeds 95% and results in inferior accuracy.

| Attack | Metric | Baseline | ATS-R | ATS-T | Soteria | Precode | DP | Top- k | Outpost | DGP |
|------------------|-----------|---------------|----------|--------|----------|--------------|---------------|--------------|--------------|--------------|
| R-gap | LPIPS | 7.7E-4 | 1.3E-4 | 0.020 | 0.378 | - | 0.373 | 0.379 | 0.378 | 0.375 |
| | SSIM | 0.965 | 0.989 | 0.870 | 0.252 | - | 0.259 | 0.249 | 0.250 | 0.248 |
| IG | LPIPS | 0.003 | 4.5E-4 | 0.108 | 0.190 | 0.371 | 0.268 | 0.029 | 0.088 | 0.316 |
| | SSIM | 0.954 | 0.981 | 0.566 | 0.368 | 0.257 | 0.333 | 0.769 | 0.640 | <u>0.287</u> |
| GI | LPIPS | 0.004 | 0.003 | 0.094 | 0.201 | 0.453 | 0.343 | 0.045 | 0.111 | 0.382 |
| | SSIM | 0.918 | 0.908 | 0.563 | 0.362 | <u>0.247</u> | 0.305 | 0.697 | 0.612 | 0.199 |
| Rob | LPIPS | 0.023 | 0.028 | 0.150 | 0.023 | 0.025 | 0.023 | <u>0.523</u> | 0.295 | 0.527 |
| | Min LPIPS | 7.43E-15 | 5.03E-15 | 0.011 | 7.79E-15 | 5.52E-15 | 8.79E-07 | <u>0.231</u> | 0.195 | 0.243 |
| | SSIM | 0.933 | 0.926 | 0.514 | 0.933 | 0.929 | 0.899 | 0.038 | 0.221 | 0.051 |
| | Max SSIM | 1.000 | 1.000 | 0.931 | 1.000 | 1.000 | 1.000 | 0.224 | <u>0.310</u> | <u>0.365</u> |
| Final Model Acc. | | 93.62% | 93.14% | 92.90% | 92.83% | 76.01% | <u>93.44%</u> | 92.96% | 93.40% | |

Table 1: Evaluation of the defense performance under four attacks.

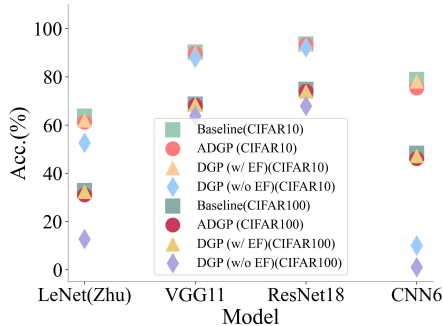


Figure 4: Evaluation of model accuracy with different datasets and models (EF denotes the error feedback).

| | $(k_1 + k_2)$ | | | $p = k_1/k_2$ | | |
|---------|---------------|-------|-------|---------------|-------|-------|
| | 48% | 80% | 96% | 1/15 | 1/7 | 1/3 |
| LPIPS | 0.426 | 0.527 | 0.531 | 0.316 | 0.351 | 0.383 |
| SSIM | 0.146 | 0.051 | 0.029 | 0.287 | 0.250 | 0.234 |
| Acc.(%) | 93.42 | 93.40 | 92.91 | 93.40 | 93.21 | 92.82 |

Table 2: The impact of different parameters on DGP.

6.4 Further Discussions

Choice of k_1 , k_2 and p for DGP. According to the analysis in Sec. 4.1, active GIA is greatly impacted by $(k_1 + k_2)$ and optimization GIA is greatly affected by $p = k_1/k_2$. In this concern, we use the Rob attack to evaluate the privacy of DGP with different $(k_1 + k_2)$ and IG attack to evaluate DGP with different p . As shown in Tab. 2, larger pruning rate $(k_1 + k_2)$ leads to better privacy-preserving, but the model’s performance suffers as a consequence. Furthermore, a larger p , i.e., more large parameters are eliminated, can better defend against optimization GIAs but impact accuracy.

Reducing download communication cost. Although DGP provides a sufficient privacy guarantee as well as reducing upload cost, users’ download cost could still be expensive. This is because different users have different sets of $\mathcal{T}_{k_1}(\cdot)$ and $\mathcal{B}_{k_2}(\cdot)$ when pruning their own local gradients, so the global model parameters will become dense after aggregation. We suggest aligned DGP (ADGP), an improved scheme to align the selected gradients to further re-

duce download cost. Similar to DGP, for best privacy, each user will still firstly identify his top- k_1 gradients location set \mathcal{T}_{k_1} . Different from DGP, ADGP also wants to save users’ download comm. cost by ensuring that all users’ uploaded pruned gradient parameters reside in the same location set. This is achieved by randomly selecting a user, who identifies a top- $2k$ ($k_1 < k$) location set \mathcal{T}_{2k} (represented with a binary location matrix \mathcal{I}) and broadcasts \mathcal{I} to all other users. Note that $\mathcal{T}_{k_1} \subset \mathcal{T}_{2k}$ is not necessarily true. Upon receiving \mathcal{I} , each user first discards gradient parameters in \mathcal{T}_{k_1} and then only transmits the k largest gradient parameters whose locations belong to \mathcal{I} . After aggregation, users only need to download the global gradients’ parameters associated with \mathcal{I} . We give the specific comm. cost in the appendix and find that ADGP further reduces the overall comm. cost. Moreover, with error feedback mechanism, it can also maintain the model performance, shown in Fig. 4. To summarize, ADGP can provide better communication efficiency while maintain model performance. We leave the work of investigating the privacy-protection of ADGP as the future work.

7 Conclusion, Limitation, and Future

Contrary to the traditional belief that gradient pruning is not a good choice to protect privacy, this paper proposes DGP, a gradient pruning-based defense, to achieve a better trade-off among privacy protection, model performance, and communication efficiency for collaborative learning. This finding is built upon the analysis of how pruned gradients bound the attacker’s recovery error and why large gradient parameters leak more private information and should be pruned. By dual-pruning both large and small gradients, DGP guarantees theoretical convergence and better privacy protection against passive attackers. By comparing to state-of-the-art defenses, experimental results corroborate our theoretical analysis, as well as empirically demonstrating the advantage of DGP against active attackers. In terms of limitations, the success of ADGP relies on selecting a reliable user to broadcast its locations. When this user becomes malicious, the entire system will fail. In the future, we will provide more rigorous and more comprehensive privacy analysis, investigate the privacy property of ADGP under passive attacks, explore the applications of (A)DGP in federated learning and broaden our research to more domains like NLP.

Acknowledgements

Shengshan’s work is supported in part by the National Natural Science Foundation of China (Grant No.U20A20177) and Hubei Province Key R&D Technology Special Innovation Project under Grant No.2021BAA032. Shengqing’s work is supported in part by Hubei Provincial Natural Science Foundation Project (NO. 2023AFB342) and Open Program of Nuclear Medicine and Molecular Imaging Key Laboratory of Hubei Province (NO. 2022fzyx018). The work is supported by HPC Platform of Huazhong University of Science and Technology. Shengshan Hu is the corresponding author.

References

- Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H. B.; Mironov, I.; Talwar, K.; and Zhang, L. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (CCS’16)*, 308–318.
- Alemi, A. A.; Fischer, I.; Dillon, J. V.; and Murphy, K. 2016. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.
- Alistarh, D.; Hoefler, T.; Johansson, M.; Konstantinov, N.; Khirirat, S.; and Renggli, C. 2018. The convergence of sparsified gradient methods. In *Proceedings of the 2018 Neural Information Processing Systems (NeurIPS’18)*, 5977–5987.
- Boenisch, F.; Dziedzic, A.; Schuster, R.; Shamsabadi, A. S.; Shumailov, I.; and Papernot, N. 2021. When the Curious Abandon Honesty: Federated Learning Is Not Private. *arXiv preprint arXiv:2112.02918*.
- Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H. B.; Patel, S.; Ramage, D.; Segal, A.; and Seth, K. 2017. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS’17)*, 1175–1191.
- Chen, C.-Y.; Ni, J.; Lu, S.; Cui, X.; Chen, P.-Y.; Sun, X.; Wang, N.; Venkataramani, S.; Srinivasan, V. V.; Zhang, W.; et al. 2020. Scalecom: Scalable sparsified gradient compression for communication-efficient distributed training. In *Proceedings of the 2020 Neural Information Processing Systems (NeurIPS’20)*, 13551–13563.
- Chen, X.; Wu, Z. S.; and Hong, M. 2020. Understanding gradient clipping in private SGD: A geometric perspective. In *Proceedings of the 2020 Neural Information Processing Systems (NeurIPS’20)*, 13773–13782.
- Dai, X.; Yan, X.; Zhou, K.; Yang, H.; Ng, K. K.; Cheng, J.; and Fan, Y. 2019. Hyper-sphere quantization: Communication-efficient sgd for federated learning. *arXiv preprint arXiv:1911.04655*.
- Danner, G.; and Jelasity, M. 2015. Fully distributed privacy preserving mini-batch gradient descent learning. In *Proceedings of the 15th International conference on distributed applications and interoperable systems (IFIP’15)*, 30–44.
- Dwork, C.; Roth, A.; et al. 2014. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4): 211–407.
- Fan, L.; Ng, K. W.; Ju, C.; Zhang, T.; Liu, C.; Chan, C. S.; and Yang, Q. 2020. Rethinking privacy preserving deep learning: How to evaluate and thwart privacy attacks. In *Federated Learning*, volume 12500, 32–50. Springer.
- Fowl, L.; Geiping, J.; Czaja, W.; Goldblum, M.; and Goldstein, T. 2021. Robbing the fed: Directly obtaining private data in federated learning with modified models. *arXiv preprint arXiv:2110.13057*.
- Gao, W.; Guo, S.; Zhang, T.; Qiu, H.; Wen, Y.; and Liu, Y. 2021. Privacy-preserving collaborative learning with automatic transformation search. In *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR’21)*, 114–123.
- Geiping, J.; Bauermeister, H.; Dröge, H.; and Moeller, M. 2020. Inverting gradients-how easy is it to break privacy in federated learning? In *Proceedings of the 2020 Neural Information Processing Systems (NeurIPS’20)*, 16937–16947.
- Geyer, R. C.; Klein, T.; and Nabi, M. 2017. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*.
- Gilad-Bachrach, R.; Laine, K.; Lauter, K.; Rindal, P.; and Rosulek, M. 2019. Secure data exchange: A marketplace in the cloud. In *Proceedings of the 2019 ACM SIGSAC Conference on Cloud Computing Security Workshop (CCSW’19)*, 117–128.
- Hardy, S.; Henecka, W.; Ivey-Law, H.; Nock, R.; Patrini, G.; Smith, G.; and Thorne, B. 2017. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR’16)*, 770–778.
- Huang, Y.; Gupta, S.; Song, Z.; Li, K.; and Arora, S. 2021. Evaluating gradient inversion attacks and defenses in federated learning. In *Proceedings of the 2021 Neural Information Processing Systems (NeurIPS’21)*, 7232–7241.
- Jeon, J.; Lee, K.; Oh, S.; Ok, J.; et al. 2021. Gradient inversion with generative image prior. In *Proceedings of the 2021 Neural Information Processing Systems (NeurIPS’21)*, 29898–29908.
- Kairouz, P.; McMahan, H. B.; Avent, B.; Bellet, A.; Bennis, M.; Bhagoji, A. N.; Bonawitz, K.; Charles, Z.; Cormode, G.; Cummings, R.; et al. 2021. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2): 1–210.
- Karimireddy, S. P.; Rebjock, Q.; Stich, S.; and Jaggi, M. 2019. Error feedback fixes signsgd and other gradient compression schemes. In *Proceedings of the 36th International Conference on Machine Learning (ICML’19)*, 3252–3261.
- Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.
- Li, Z.; Zhang, J.; Liu, L.; and Liu, J. 2022. Auditing Privacy Defenses in Federated Learning via Generative Gradient Leakage. In *Proceedings of the 2022 IEEE/CVF Conference*

- on *Computer Vision and Pattern Recognition (CVPR'22)*, 10132–10142.
- Lin, Y.; Han, S.; Mao, H.; Wang, Y.; and Dally, W. J. 2017. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887*.
- Liu, R.; Cao, Y.; Chen, H.; Guo, R.; and Yoshikawa, M. 2020. Flame: Differentially private federated learning in the shuffle model. *arXiv preprint arXiv:2009.08063*.
- Mohassel, P.; and Zhang, Y. 2017. Secureml: A system for scalable privacy-preserving machine learning. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy (SP'17)*, 19–38.
- Molchanov, P.; Tyree, S.; Karras, T.; Aila, T.; and Kautz, J. 2016. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*.
- Pan, X.; Zhang, M.; Yan, Y.; Zhu, J.; and Yang, Z. 2022. Exploring the security boundary of data reconstruction via neuron exclusivity analysis. In *Proceedings of the 31st USENIX Security Symposium (Security'22)*, 3989–4006.
- Phong, L. T.; Aono, Y.; Hayashi, T.; Wang, L.; and Moriai, S. 2017. Privacy-preserving deep learning: Revisited and enhanced. In *Proceedings of the 8th International Conference on Applications and Techniques in Information Security (ATIS'17)*, 100–110.
- Qian, J.; and Hansen, L. K. 2020. What can we learn from gradients? *arXiv preprint arXiv:2010.15718*.
- Scheliga, D.; Mäder, P.; and Seeland, M. 2022. PRECODE-A Generic Model Extension to Prevent Deep Gradient Leakage. In *Proceedings of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV'22)*, 1849–1858.
- Shokri, R.; and Shmatikov, V. 2015. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security (CCS'15)*, 1310–1321.
- Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Sun, J.; Li, A.; Wang, B.; Yang, H.; Li, H.; and Chen, Y. 2021. Soteria: Provable defense against privacy leakage in federated learning from representation perspective. In *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*, 9311–9319.
- Sun, L.; Qian, J.; and Chen, X. 2021. Ldp-fl: Practical private aggregation in federated learning with local differential privacy. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI'21)*, 1571–1578.
- Wang, F.; Hugh, E.; and Li, B. 2023. More than Enough is Too Much: Adaptive Defenses against Gradient Leakage in Production Federated Learning. In *Proceedings of the International Conference on Computer Communications (Infocom'23)*.
- Wang, Y.; Deng, J.; Guo, D.; Wang, C.; Meng, X.; Liu, H.; Ding, C.; and Rajasekaran, S. 2020. Sapag: A self-adaptive privacy attack from gradients. *arXiv preprint arXiv:2009.06228*.
- Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.
- Wang, Z.; Song, M.; Zhang, Z.; Song, Y.; Wang, Q.; and Qi, H. 2019. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *Proceedings of the 2019 IEEE Conference on Computer Communications (INFOCOM'19)*, 2512–2520.
- Wei, K.; Li, J.; Ding, M.; Ma, C.; Yang, H. H.; Farokhi, F.; Jin, S.; Quek, T. Q.; and Poor, H. V. 2020a. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15: 3454–3469.
- Wei, W.; Liu, L.; Loper, M.; Chow, K.-H.; Gursoy, M. E.; Truex, S.; and Wu, Y. 2020b. A framework for evaluating gradient leakage attacks in federated learning. *arXiv preprint arXiv:2004.10397*.
- Wen, Y.; Geiping, J.; Fowl, L.; Goldblum, M.; and Goldstein, T. 2022. Fishing for User Data in Large-Batch Federated Learning via Gradient Magnification. *arXiv preprint arXiv:2202.00580*.
- Wilson, A. C.; Roelofs, R.; Stern, M.; Srebro, N.; and Recht, B. 2017. The marginal value of adaptive gradient methods in machine learning. In *Proceedings of the 2017 Neural Information Processing Systems (NeurIPS'17)*, 4148–4158.
- Yin, H.; Mallya, A.; Vahdat, A.; Alvarez, J. M.; Kautz, J.; and Molchanov, P. 2021. See through gradients: Image batch recovery via gradinversion. In *Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR'21)*, 16337–16346.
- Yin, H.; Molchanov, P.; Alvarez, J. M.; Li, Z.; Mallya, A.; Hoiem, D.; Jha, N. K.; and Kautz, J. 2020. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8715–8724.
- Yu, L.; Liu, L.; Pu, C.; Gursoy, M. E.; and Truex, S. 2019. Differentially private model publishing for deep learning. In *Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP'19)*, 332–349.
- Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the 2018 IEEE conference on computer vision and pattern recognition (CVPR'18)*, 586–595.
- Zhao, B.; Mopuri, K. R.; and Bilen, H. 2020. idlg: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610*.
- Zhu, J.; and Blaschko, M. 2020. R-gap: Recursive gradient attack on privacy. *arXiv preprint arXiv:2010.07733*.
- Zhu, L.; Liu, Z.; and Han, S. 2019. Deep leakage from gradients. In *Proceedings of the 2019 Neural Information Processing Systems (NeurIPS'19)*, 14747–14756.

A Theoretical proof

This section presents all the missing theoretical analyses appeared in the manuscript orderly.

Proposition 2. For any given input \mathbf{x} and shared model \mathbf{W} , the distance between the recovered data \mathbf{x}' and the real data \mathbf{x} is bounded by:

$$\|\mathbf{x} - \mathbf{x}'\|_2 \geq \frac{\|\varphi(\mathbf{x}, \mathbf{W}) - \varphi(\mathbf{x}', \mathbf{W})\|_2}{\|\partial\varphi(\mathbf{x}, \mathbf{W})/\partial\mathbf{x}\|_2},$$

Proof. Apply the first-order Taylor expansion to $(\varphi(\mathbf{x}, \mathbf{W}) - \varphi(\mathbf{x}', \mathbf{W}))$, it is easy to find

$$\begin{aligned} & \|\varphi(\mathbf{x}, \mathbf{W}) - \varphi(\mathbf{x}', \mathbf{W})\|_2 \\ & \approx \|(\partial\varphi(\mathbf{x}, \mathbf{W})/\partial\mathbf{x})(\mathbf{x} - \mathbf{x}')\|_2 \\ & \leq \|(\partial\varphi(\mathbf{x}, \mathbf{W})/\partial\mathbf{x})\|_2 \|\mathbf{x} - \mathbf{x}'\|_2. \end{aligned}$$

Hence, we have

$$\|\mathbf{x} - \mathbf{x}'\|_2 \geq \frac{\|\varphi(\mathbf{x}, \mathbf{W}) - \varphi(\mathbf{x}', \mathbf{W})\|_2}{\|\partial\varphi(\mathbf{x}, \mathbf{W})/\partial\mathbf{x}\|_2}. \quad (5)$$

□

Theorem 3. For any (ε, δ) -passive attack \mathcal{A} , under the presence of DGP, it will be degenerated to $(\varepsilon + \sqrt{\gamma_1}\|\nabla\mathbf{W}\|_2, \delta)$ -passive attack if $\mathcal{D}_{\mathcal{A}}$ is measured by Euclidean distance, and degenerated to $(\varepsilon + (1 - \varepsilon)\sqrt{\gamma_1}, \delta)$ -passive attack if $\mathcal{D}_{\mathcal{A}}$ is measured by cosine distance.

Proof. If $\mathcal{D}_{\mathcal{A}}$ is measured by Euclidean distance, by the definition of (ε, δ) -attack, the attacker can achieve the following estimation

$$\mathbb{E}\|\mathbf{g}^* - \nabla\mathbf{W}\|_2 \leq \varepsilon,$$

where \mathbf{g}^* is the attacker's optimized gradients of the ground-truth gradients \mathbf{W} . When DGP is used, from the bi-Lipschitz assumption (i.e., Assumption 1), we know

$$\|\text{DGP}(\nabla\mathbf{W}) - \nabla\mathbf{W}\|_2 \leq \sqrt{\gamma_1}\|\nabla\mathbf{W}\|_2. \quad (6)$$

Then, when central aggregation is protected by DGP, the attacker's optimized gradients is based on the observation of $\text{DGP}(\nabla\mathbf{W})$ and this modified observation will degrade the attacker's capability in optimizing $\nabla\mathbf{W}$ because

$$\begin{aligned} & \mathbb{E}\|\mathbf{g}^* - \nabla\mathbf{W}\|_2 \\ & = \mathbb{E}\|\mathbf{g}^* - \text{DGP}(\nabla\mathbf{W}) + \text{DGP}(\nabla\mathbf{W}) - \nabla\mathbf{W}\|_2 \\ & \leq \varepsilon + \|\text{DGP}(\nabla\mathbf{W}) - \nabla\mathbf{W}\|_2 \\ & \leq \varepsilon + \sqrt{\gamma_1}\|\nabla\mathbf{W}\|_2. \end{aligned}$$

Hence, the first part of this theorem is true.

Similarly, when $\mathcal{D}_{\mathcal{A}}$ is measured by cosine distance, the definition of (ε, δ) -attack reveals

$$\mathbb{E}\left[1 - \frac{\langle \mathbf{g}^*, \nabla\mathbf{W} \rangle}{\|\mathbf{g}^*\|_2 \|\nabla\mathbf{W}\|_2}\right] \leq \varepsilon.$$

Then, we can obtain

$$\begin{aligned} & \mathbb{E}\left[\frac{\langle \mathbf{g}^*, \nabla\mathbf{W} \rangle}{\|\mathbf{g}^*\|_2 \|\nabla\mathbf{W}\|_2}\right] \\ & = \mathbb{E}\left[\frac{\langle \mathbf{g}^*, \nabla\mathbf{W} - \text{DGP}(\nabla\mathbf{W}) + \text{DGP}(\nabla\mathbf{W}) \rangle}{\|\mathbf{g}^*\|_2 \|\nabla\mathbf{W}\|_2}\right] \\ & \stackrel{(c)}{=} \mathbb{E}\left[\frac{\langle \mathbf{g}^*, \text{DGP}(\nabla\mathbf{W}) \rangle}{\|\mathbf{g}^*\|_2 \|\nabla\mathbf{W}\|_2}\right] \\ & = \mathbb{E}\left[\frac{\langle \mathbf{g}^*, \text{DGP}(\nabla\mathbf{W}) \rangle}{\|\mathbf{g}^*\|_2 \|\text{DGP}(\nabla\mathbf{W})\|_2} \frac{\|\text{DGP}(\nabla\mathbf{W})\|_2}{\|\nabla\mathbf{W}\|_2}\right] \\ & \stackrel{(d)}{\geq} (1 - \sqrt{\gamma_1})\mathbb{E}\left[\frac{\langle \mathbf{g}^*, \text{DGP}(\nabla\mathbf{W}) \rangle}{\|\mathbf{g}^*\|_2 \|\text{DGP}(\nabla\mathbf{W})\|_2}\right] \\ & \geq (1 - \sqrt{\gamma_1})(1 - \varepsilon) = 1 + \varepsilon\sqrt{\gamma_1} - \sqrt{\gamma_1} - \varepsilon. \quad (7) \end{aligned}$$

where (c) is based on the fact that the all non-zero elements of $(\nabla\mathbf{W} - \text{DGP}(\nabla\mathbf{W}))$ are pruned in DGP so $\mathbb{E}(\mathbf{g}^*, (\nabla\mathbf{W} - \text{DGP}(\nabla\mathbf{W}))) = 0$, and (d) is the direct application of Eq. (6). Based on Eq. (7), it is easy to conclude

$$\mathbb{E}\left[1 - \frac{\langle \mathbf{g}^*, \nabla\mathbf{W} \rangle}{\|\mathbf{g}^*\|_2 \|\nabla\mathbf{W}\|_2}\right] \leq \varepsilon + (1 - \varepsilon)\sqrt{\gamma_1},$$

which completes the proof. □

Lemma 2. Let $\mathbf{e}_t = \sum_{i=1}^N \mathbf{e}_{t,i}/N$ be the averaged accumulated error among all users at iteration t , the expectation of the norm of \mathbf{e}_t is bounded, i.e.,

$$\mathbb{E}\|\mathbf{e}_t\|_2^2 \leq \frac{3\gamma_1(2 + \gamma_1)}{2(1 - \gamma_1)^2} G^2.$$

Proof. To use the theoretical tools of SGD, we set up the following dummy matrix \mathbf{V} :

$$\mathbf{V}_{t+1} = \mathbf{V}_t - \eta\nabla\mathbf{W}_t.$$

Since $\mathbf{W}^0 = \mathbf{V}^0$, $\mathbf{e}^0 = 0$, it is easy to find

$$\mathbf{V}_t - \mathbf{W}_t = \eta\mathbf{e}_t. \quad (8)$$

Under Assumption 1, we have

$$\|\mathbf{X} - \text{DGP}(\mathbf{X})\|_2^2 \leq \gamma_1\|\mathbf{X}\|_2^2 \quad (9)$$

Under Assumption 3, we have

$$\mathbb{E}\|\nabla\mathbf{W}_{t,i}\|_2^2 \leq G^2 \quad (10)$$

$$\mathbb{E}\|\nabla\mathbf{W}_t\|_2^2 \leq G^2 + \sigma^2. \quad (11)$$

By definition of \mathbf{e}_t , we know

$$\|\mathbf{e}_t\|_2^2 \leq \frac{\sum_{i=1}^N \|\mathbf{e}_{t,i}\|_2^2}{N},$$

and the $\|\mathbf{e}_{t,i}\|_2^2$ is also bounded because

$$\begin{aligned} \|\mathbf{e}_{t,i}\|_2^2 & = \|\nabla\mathbf{W}_{t-1,i} + \mathbf{e}_{t-1,i} - \text{DGP}(\nabla\mathbf{W}_{t-1,i} + \mathbf{e}_{t-1,i})\|_2^2 \\ & \stackrel{(9)}{\leq} \gamma_1\|\nabla\mathbf{W}_{t-1,i} + \mathbf{e}_{t-1,i}\|_2^2 \\ & \stackrel{(e)}{\leq} \gamma_1\left(\left(1 + \frac{1}{a}\right)\|\nabla\mathbf{W}_{t-1,i}\|_2^2 + (1 + a)\|\mathbf{e}_{t-1,i}\|_2^2\right). \end{aligned}$$

where (e) is based on the variant of Young's inequality $\|x + y\|_2^2 \leq (1 + a)\|x\|_2^2 + (1 + \frac{1}{a})\|y\|_2^2$. Set $1 + a = \frac{2+\gamma_1}{3\gamma_1}$, it is concluded that

$$\mathbb{E}\|\mathbf{e}_{t,i}\|_2^2 \stackrel{(10)}{\leq} \frac{3\gamma_1(2+\gamma_1)}{2(1-\gamma_1)^2} G^2, \quad (12)$$

$$\mathbb{E}\|\mathbf{e}_t\|_2^2 \leq \frac{3\gamma_1(2+\gamma_1)}{2(1-\gamma_1)^2} G^2. \quad (13)$$

□

Theorem 4. *The averaged norm of the full gradient $\nabla l(\mathbf{W}_t)$ derived from centralized training is correlated with the our algorithm as follows:*

$$\frac{\sum_{t=0}^{T-1} \mathbb{E}\|\nabla l(\mathbf{W}_t)\|_2^2}{T} \leq 4\frac{K^0 - l^*}{\eta T} + 4\eta^2 K^2 \frac{3\gamma_1(2+\gamma_1)}{2(1-\gamma_1)^2} G^2 + 2K\eta(G^2 + \sigma^2), \quad (14)$$

where l^0 is the initialization of the objective l , and η is the learning rate.

Proof. Under Assumption 3, we have

$$\|\nabla l(\mathbf{V}_t) - \nabla l(\mathbf{W}_t)\| \leq K\|\mathbf{V}_t - \mathbf{W}_t\|, \quad (15)$$

and

$$\begin{aligned} l(\mathbf{V}_{t+1}) &\leq l(\mathbf{V}_t) + \langle \nabla l(\mathbf{V}_t), \mathbf{V}_{t+1} - \mathbf{V}_t \rangle + \frac{K}{2}\|\mathbf{V}_{t+1} - \mathbf{V}_t\|_2^2 \\ &= l(\mathbf{V}_t) - \eta \langle \nabla l(\mathbf{V}_t), \nabla \mathbf{W}_t \rangle + \frac{K\eta^2}{2}\|\nabla \mathbf{W}_t\|_2^2. \end{aligned} \quad (16)$$

Taking expectation on both sides of Eq. (16), we can get

$$\begin{aligned} \mathbb{E} l(\mathbf{V}_{t+1}) &\leq \mathbb{E}(l(\mathbf{V}_t)) - \eta \mathbb{E}(\langle \nabla l(\mathbf{V}_t), \nabla l(\mathbf{W}_t) \rangle) \\ &+ \frac{K\eta^2}{2} \mathbb{E}\|\nabla \mathbf{W}_t\|_2^2 \\ &\stackrel{(f)}{=} \mathbb{E}(l(\mathbf{V}_t)) - \frac{\eta}{2} (\mathbb{E}(\|\nabla l(\mathbf{V}_t)\|_2^2) + \mathbb{E}(\|\nabla l(\mathbf{W}_t)\|_2^2)) \\ &+ \frac{\eta}{2} \mathbb{E}\|\nabla l(\mathbf{V}_t) - \nabla l(\mathbf{W}_t)\|_2^2 + \frac{K\eta^2}{2} \mathbb{E}\|\nabla \mathbf{W}_t\|_2^2 \\ &\leq \mathbb{E}(l(\mathbf{V}_t)) - \frac{\eta}{2} \mathbb{E}(\|\nabla l(\mathbf{V}_t)\|_2^2) + \frac{K\eta^2}{2} \mathbb{E}\|\nabla \mathbf{W}_t\|_2^2 \\ &+ \frac{\eta}{2} \mathbb{E}\|\nabla l(\mathbf{V}_t) - \nabla l(\mathbf{W}_t)\|_2^2 \\ &\stackrel{(15)}{\leq} \mathbb{E}(l(\mathbf{V}_t)) - \frac{\eta}{2} (\mathbb{E}\|\nabla l(\mathbf{V}_t)\|_2^2) + \frac{K\eta^2}{2} \mathbb{E}\|\nabla \mathbf{W}_t\|_2^2 \\ &+ \frac{K^2\eta}{2} \mathbb{E}\|\mathbf{V}_t - \mathbf{W}_t\|_2^2 \\ &\stackrel{(8)}{\leq} \mathbb{E}(l(\mathbf{V}_t)) - \frac{\eta}{2} (\mathbb{E}\|\nabla l(\mathbf{V}_t)\|_2^2) + \frac{\eta^3 K^2}{2} \mathbb{E}\|\mathbf{e}_t\|_2^2 \\ &+ \frac{K\eta^2}{2} \mathbb{E}\|\nabla \mathbf{W}_t\|_2^2 \\ &\stackrel{(11)}{\leq} \mathbb{E}(l(\mathbf{V}_t)) - \frac{\eta}{2} (\mathbb{E}\|\nabla l(\mathbf{V}_t)\|_2^2) + \frac{\eta^3 K^2}{2} \mathbb{E}\|\mathbf{e}_t\|_2^2 \\ &+ \frac{K\eta^2}{2} (G^2 + \sigma^2), \end{aligned} \quad (17)$$

where (f) is based on the fact $\langle x, y \rangle = \frac{1}{2}(\|x\|^2 + \|y\|^2 - \|x - y\|^2)$. Base on the deduction above, we can further calculate

$$\begin{aligned} \frac{\eta}{2} (\mathbb{E}\|\nabla l(\mathbf{V}_t)\|_2^2) &\leq \mathbb{E}(l(\mathbf{V}_t)) - \mathbb{E}(l(\mathbf{V}_{t+1})) + \frac{\eta^3 K^2}{2} \mathbb{E}\|\mathbf{e}_t\|_2^2 \\ &+ \frac{K\eta^2}{2} (G^2 + \sigma^2), \end{aligned} \quad (18)$$

$$\begin{aligned} \left(\frac{\sum_0^{T-1} \mathbb{E}\|\nabla l(\mathbf{V}_t)\|_2^2}{T} \right) &\leq \frac{2(l^0 - l^*)}{\eta T} + \eta^2 K^2 \mathbb{E}\|\mathbf{e}_t\|_2^2 \\ &+ K\eta(G^2 + \sigma^2). \end{aligned} \quad (19)$$

According to Eq. (15), it can be found that

$$\begin{aligned} \|\nabla l(\mathbf{W}_t)\| &\leq K\|\mathbf{V}_t - \mathbf{W}_t\| + \|\nabla l(\mathbf{V}_t)\|, \\ \|\nabla l(\mathbf{W}_t)\|_2^2 &\leq 2K^2\|\mathbf{V}_t - \mathbf{W}_t\|_2^2 + 2\|\nabla l(\mathbf{V}_t)\|_2^2. \end{aligned} \quad (20)$$

Combining Eq. (13), Eq. (19) and Eq. (20), it is concluded

$$\begin{aligned} \mathbb{E}\|\nabla l(\mathbf{W}_t)\|_2^2 &\leq \frac{4(l^0 - l^*)}{\eta T} + 4\eta^2 K^2 \mathbb{E}\|\mathbf{e}_t\|_2^2 \\ &+ 2K\eta(G^2 + \sigma^2) \\ &\leq \frac{4(l^0 - l^*)}{\eta T} + 4\eta^2 K^2 \frac{3\gamma_1(2+\gamma_1)}{2(1-\gamma_1)^2} G^2 + 2K\eta(G^2 + \sigma^2). \end{aligned}$$

Set $\eta = \sqrt{\frac{l^0 - l^*}{KT(\sigma^2 + G^2)}}$, we have

$$\frac{\sum_0^{T-1} \mathbb{E}\|\nabla l(\mathbf{W}_t)\|_2^2}{T} \leq 6\sqrt{\frac{K(l^0 - l^*)(\sigma^2 + G^2)}{T}} + \mathcal{O}\left(\frac{1}{T}\right).$$

Hence, the theorem is true. □

B Analysis of Assumption

Since Assumption 2 and Assumption 3 are common assumptions in many works (Karimireddy et al. 2019; Wilson et al. 2017), this section focus on analyzing the feasibility of Assumption 1.

Assumption 4. *The pruning mechanism $\text{DGP}(k_1, k_2, \cdot)$ is Lipschitz, so the following conditions hold:*

$$\begin{aligned} \|\nabla \mathbf{W} - \text{DGP}(k_1, k_2, \nabla \mathbf{W})\|_2^2 \\ = \|\text{DGP}(0, 0, \nabla \mathbf{W}) - \text{DGP}(k_1, k_2, \nabla \mathbf{W})\|_2^2 &\leq \gamma_1 \|\nabla \mathbf{W}\|_2^2, \end{aligned}$$

where γ_1 is a constant related to k_1 and k_2 and satisfies $(1 - \sqrt{1 - k_1 * k_2})^2 < \gamma_1 < 1$.

To simplify the expression, we use $\text{DGP}(\nabla \mathbf{W})$ to denote $\text{DGP}(\nabla \mathbf{W}, k_1, k_2)$ and $\|\cdot\|$ to denote $\|\cdot\|_2$. (Alistarh et al. 2018) states the following property of $\text{top}[l](\nabla \mathbf{W})$ (i.e., retain the top l -ratio of $\nabla \mathbf{W}$):

$$\|\nabla \mathbf{W} - \text{top}[l](\nabla \mathbf{W})\| \leq \sqrt{1 - l} \|\nabla \mathbf{W}\|. \quad (21)$$

According to formula 21, it is easy to obtain formula 22:

$$\|\text{top}[l](\nabla \mathbf{W})\| \geq (1 - \sqrt{1 - l}) \|\nabla \mathbf{W}\|. \quad (22)$$

And easy to find:

$$\begin{aligned} \|\nabla \mathbf{W} - \text{DGP}(k_1, k_2, \nabla \mathbf{W})\|_2 \\ = \|\text{top}[k_1](\nabla \mathbf{W}) + \text{bottom}[k_2](\nabla \mathbf{W})\|_2 \\ > \|\text{top}[k_1 * k_2](\nabla \mathbf{W})\|_2. \end{aligned}$$

Combined with formula 22, it is true that $\gamma_1 > (1 - \sqrt{1 - k_1 * k_2})^2$. Moreover, even if the difference is large, i.e., almost all parameters are removed and γ_1 approaches 1, the assumption still holds.

C More experimental results

This section presents the related experimental setup and more experimental results.

C.1 Experimental setup

Privacy experiment setup. About IG and GI, follow (Gao et al. 2021), we set the number of iterations to 2500, and we set the optimal learning rate to 0.1. For rob attack, we set the attack batchsize=9. For IG, GI and R-GAP attacks, we set attack batchsize=1, randomly select 20 data points for attack, and calculate the average metrics of the attack results. In addition, there are some settings about defenses. We set the pruning rate of Soteria to 80, set the Outpost hyperparameters as $\lambda=0.8$, $\varphi=40$, $\beta=0.1$, $\rho=80$, which are the originally experimental setting. In addition, we configured an LPIPS object employing AlexNet as the perceptual model, considering the spatial structure of images for comparison.

Accuracy experiment setup. We train models with batchsize=32. We use SGD optimizer with momentum of 0.9 and set epoch=100. To ensure a good performance of the baseline, we set the following learning rates. For LeNet (Zhu), we set the learning rate $\eta=0.1$ if epoch ≤ 50 , $\eta=0.01$ if epoch >50 , and $\eta=0.005$ if epoch >70 . For the rest of the training settings, we set the learning rate $\eta=0.01$ if epoch ≤ 70 , $\eta=0.005$ if epoch >70 . Considering the error feedback is designed for improving model accuracy for gradient pruning (Karimireddy et al. 2019), we also apply error feedback to Top- k .

C.2 Efficiency Evaluation

Communication cost. We measure the communication cost of the defenses for all trained models and list the average result of one epoch in Tab. 3. DP and ATS do not affect the number of transmitted parameters, their results are the same as the baseline. Clearly, DGP and Top- k save about 40% bandwidth when comparing to the baseline. Outpost perturbs the gradient parameters based on Top- k pruning, so the result is consistent with Top- k . Note that this advantage is free since gradient pruning incurs a negligible computation burden (detailed evaluation is presented in the Tab. 4).

| Model | Baseline | Soteria | Precode | Top- k | DGP | ADGP |
|-------------|----------|---------|---------|----------|--------|---------------|
| LeNet (Zhu) | 0.121 | 0.098 | 4.631 | 0.074 | 0.074 | 0.038 |
| VGG11 | 70.428 | 70.413 | 73.436 | 43.137 | 43.137 | 22.449 |
| ResNet18 | 85.251 | 85.235 | 88.258 | 52.216 | 52.216 | 27.174 |
| CNN6 | 1.177 | 0.959 | 19.955 | 0.721 | 0.721 | 0.375 |

Table 3: Average overall comm. cost in one epoch (MB).

Computation cost. Tab. 4 shows the computation cost comparison of gradient parameter searching for one epoch. Although the average computation cost of ADGP is slightly higher than DGP because ADGP needs to load the binary matrix \mathcal{I} , this computation cost is trivial considering the reduced communication cost. And our method is obviously better than Soteria, because Soteria requires a lot of computation on gradients, which leads to expensive computation cost.

| | Soteria | Top- k | DGP | ADGP |
|-------------|---------|----------|-------|-------|
| LeNet (Zhu) | 52.460 | 0.113 | 0.146 | 0.167 |
| VGG11 | 331.379 | 0.419 | 0.764 | 0.842 |
| ResNet18 | 862.567 | 0.967 | 1.803 | 2.587 |
| CNN6 | 291.419 | 0.100 | 0.237 | 0.256 |

Table 4: Average comp. cost in one epoch (s)

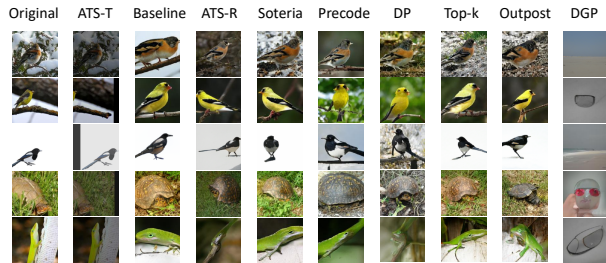


Figure 5: Reconstruction data visualization under GGL attack on ImageNet.

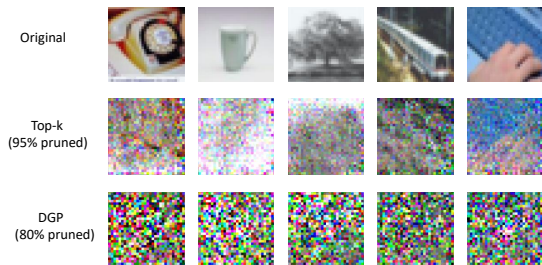


Figure 6: Reconstruction data visualization under IG attack on CIFAR100.

C.3 Privacy evaluation with more models.

In this section, We evaluate privacy with multiple models and datasets. We choose the state-of-the-art active attack Rob and the state-of-the-art passive attack IG for privacy evaluation. As shown in Tab. 5 and Tab. 6, DGP can play an effective privacy protection.

C.4 Defenses under attacks with generative GIAs.

For a comprehensive privacy evaluation, we assess existing defenses against GAN-based generative GIAs. We select GGL, the state-of-the-art generative GIA, and maintain its original strongest configuration, utilizing ResNet18 on ImageNet. Fig. 5 shows that DGP has a better visualization effect. This is because pruning Top- k_1 gradient elements (in DGP) will confuse GGL’s inference of some data labels, making the GAN-generated relevant data differ from the original data significantly.

C.5 Evaluation of the trade-off between privacy and accuracy for high-pruning rate Top- k

Set Top- k pruning rate for 95%, DGP pruning rate for 80%, we compare their privacy-accuracy trade-offs. As shown in Fig. 6, Fig. 7 and Tab. 7, with similar privacy protection, Top- k is more likely to cause model performance degradation.

| | | CNN6 | | LeNet (Zhu) | | ResNet18 | | VGG11 | |
|------------|--------|----------|---------------|-------------|---------------|----------|---------------|----------|---------------|
| Attack | Metric | Baseline | DGP | Baseline | DGP | Baseline | DGP | Baseline | DGP |
| Rob attack | LPIPS | 0.0216 | 0.4421 | 0.0226 | 0.3031 | 0.0231 | 0.5273 | 0.0272 | 0.5079 |
| | SSIM | 0.9273 | 0.1335 | 0.9293 | 0.2364 | 0.9328 | 0.0511 | 0.9328 | 0.0464 |
| IG attack | LPIPS | 0.0308 | 0.2349 | 0.0788 | 0.2537 | 0.0028 | 0.3163 | 0.0303 | 0.2845 |
| | SSIM | 0.7735 | 0.4375 | 0.6745 | 0.3785 | 0.9539 | 0.2866 | 0.8084 | 0.3269 |

Table 5: Privacy Evaluation on CIFAR10.

| | | CNN6 | | LeNet (Zhu) | | ResNet18 | | VGG11 | |
|------------|--------|----------|---------------|-------------|---------------|----------|---------------|----------|---------------|
| Attack | Metric | Baseline | DGP | Baseline | DGP | Baseline | DGP | Baseline | DGP |
| Rob attack | LPIPS | 0.0212 | 0.4726 | 0.0322 | 0.3782 | 0.0297 | 0.4643 | 0.0248 | 0.4677 |
| | SSIM | 0.9298 | 0.1177 | 0.9273 | 0.2257 | 0.9322 | 0.1474 | 0.9157 | 0.1209 |
| IG attack | LPIPS | 0.0521 | 0.2996 | 0.1042 | 0.3163 | 0.0021 | 0.3747 | 0.0297 | 0.3237 |
| | SSIM | 0.7551 | 0.3792 | 0.6849 | 0.3781 | 0.9522 | 0.2550 | 0.8057 | 0.2991 |

Table 6: Privacy Evaluation on CIFAR100.

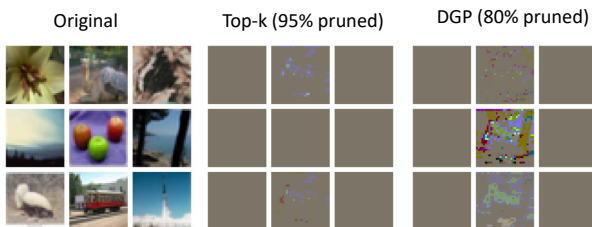
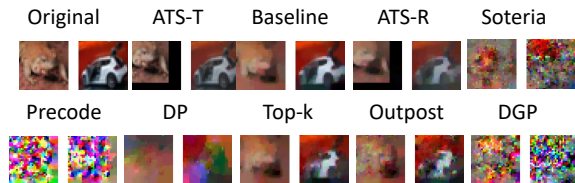
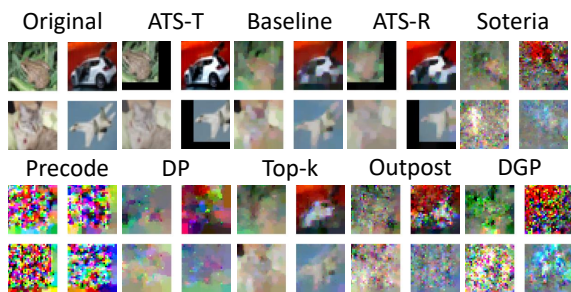


Figure 7: Reconstruction data visualization under Rob attack on CIFAR100.



(a) IG attack, batchsize=2



(b) IG attack, batchsize=4

Figure 8: Reconstruction data visualization under IG attack with different batchsizes on CIFAR10.

C.6 Defenses under attacks with different batch sizes

To better evaluate privacy protection, we implement IG attack and Rob attack with different batchsizes. Fig. 8 and

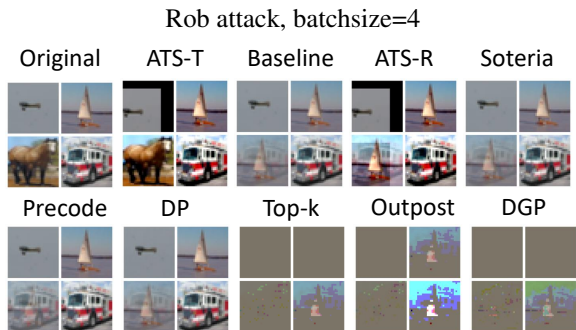
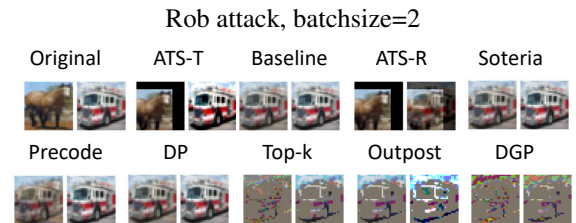


Figure 9: Reconstruction data visualization under Rob attack with different batchsizes on CIFAR10.

| | ResNet18 | LeNet (Zhu) | CNN6 | VGG11 |
|-------------|---------------|---------------|---------------|---------------|
| Top-k (95%) | 73.68% | 26.75% | 45.19% | 68.07% |
| DGP (80%) | 74.04% | 32.42% | 47.24% | 68.60% |

Table 7: Model performance on CIFAR100.

Fig. 9 show that our method protect privacy against IG and Rob attacks better than recent works. In particular, our method can comprehensively defend against gradient inversion attacks, while Top-k and Outpost offer limited privacy protection against IG attack, and Soteria, ATS, Precode cannot defend against Rob attack.

D The framework of ADGP

As shown in Fig. 10, ADGP is achieved by randomly selecting a user, who broadcasts binary matrix \mathcal{L} to all other users. Each user then only transmits gradient parameters whose lo-

cations belong to \mathcal{I} .

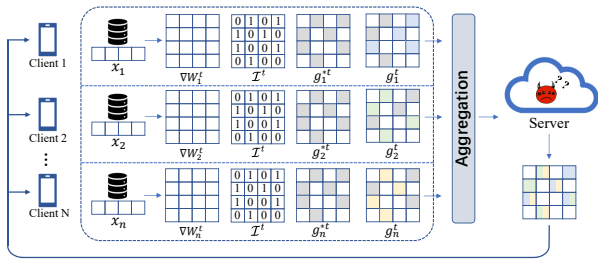


Figure 10: The t -th iteration model update process, where g^{*t} represents the gradient parameters whose position belong to \mathcal{I} in t -th iteration.